

1 SECURE INFORMATION DISTRIBUTION BETWEEN
2 NODES (NETWORK DEVICES)

3
4 Inventor(s): Michael Roeder
5 Ponnappa Palecanda
6

7
8 TECHNICAL FIELD

9 Embodiments of the present invention relate generally
10 to communication networks, and more particularly to the
11 distribution of secure information between network devices.
12

13 BACKGROUND

14 Current methods in the administration of network
15 passwords and distribution of key information are time
16 consuming and complicated. For example, network managers
17 are required to manually program the passwords, key
18 information, and/or other secure information into each
19 network device in a network, when the password, key
20 information, and/or other secure information are updated.

21 As another example of a current method, an
22 authentication server in the network is used and is queried
23 by the network device for the passwords or key information.
24 However, the authentication server may be disadvantageously
25 subjected to network failures such as link failures and
26 server device failures. As such, a network failure will

1 not permit other network device to obtain the updated
2 passwords or key information or other important secure
3 information.

4 Therefore, the current technology is limited in its
5 capabilities and suffers from at least the above
6 constraints or deficiencies.

1 SUMMARY OF EMBODIMENTS OF THE INVENTION

2 In one embodiment of the invention, a method of secure
3 information distribution between nodes, includes:
4 performing a handshake process with an adjacent node to
5 determine membership in a secure group; and distributing
6 secure information to the adjacent node, if the adjacent
7 node is a member of the secure group.

8 In another embodiment of the invention, an apparatus
9 for secure information distribution between nodes,
10 includes: a node configured to performing a handshake
11 process with an adjacent node to determine membership in a
12 secure group, and distribute secure information to the
13 adjacent node, if the adjacent node is a member of the
14 secure group.

15 These and other features of embodiments of the
16 invention will be readily apparent to persons of ordinary
17 skill in the art upon reading the entirety of this
18 disclosure, which includes the accompanying drawings and
19 claims.

1 BRIEF DESCRIPTION OF THE DRAWINGS

2 Non-limiting and non-exhaustive embodiments of the
3 present invention are described with reference to the
4 following figures, wherein like reference numerals refer to
5 like parts throughout the various views unless otherwise
6 specified.

7

8 Figure 1 is a block diagram of a system (apparatus)
9 that can implement an embodiment of the invention.

10 Figure 2 is a flowchart of a method, in accordance
11 with an embodiment of the invention.

12 Figure 3 is a block diagram of a table with secure
13 information, in accordance with an embodiment of the
14 invention.

15 Figure 4 is a block diagram shown for the purpose of
16 illustrating a method to resolve ambiguity between entry
17 updates.

18 Figure 5 is a block diagram illustrating a method for
19 increasing the security of the secure group.

1 DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

2 In the description herein, numerous specific details
3 are provided, such as examples of components and/or
4 methods, to provide a thorough understanding of embodiments
5 of the invention. One skilled in the relevant art will
6 recognize, however, that an embodiment of the invention can
7 be practiced without one or more of the specific details,
8 or with other apparatus, systems, methods, components,
9 materials, parts, and/or the like. In other instances,
10 well-known structures, materials, or operations are not
11 shown or described in detail to avoid obscuring aspects of
12 embodiments the invention.

13

14 Figure 1 is a block diagram of a system (apparatus)
15 100 that can implement an embodiment of the invention. The
16 system 100 is implemented in a communication network, and
17 is used to distribute sensitive or secure information
18 between sets of nodes (network devices). The nodes are
19 generally referred to herein as nodes 105. In the example
20 of Figure 1, nodes 105a, 105b, and 105c are shown for
21 purposes of describing the operation of embodiments of the
22 invention.

23 In an embodiment the node 105a includes the following
24 features or elements. It is noted that the other nodes

1 105b and 105c includes similar features or elements. A
2 management module 110 can query, read, and write data to
3 data structures in a database 115. An inbound packet
4 system 120 receives and processes incoming packets. The
5 inbound packet system 120 typically includes a time stamp
6 module 125 that places a timestamp on stored data in the
7 database 115.

8 A hello packet process 130 is configured to receive,
9 process, and acknowledge the incoming HELLO packets. The
10 hello packet process 130 receives the HELLO packets and
11 detects nodes 105 that are not currently in an adjacency
12 set of the node 105a. The node 105a will then attempt to
13 handshake with those detected nodes. As known to those
14 skilled in the art, in the Open Shortest Path First (OSPF)
15 communications protocol (which enables network routers to
16 share information with each other), a HELLO packet is a
17 special packet (message) that is sent out periodically from
18 a router to establish and confirm network adjacency
19 relationships. On networks capable of broadcast or
20 multicast transmission, a HELLO packet can be sent from a
21 router simultaneously to other routers to discover
22 neighboring routers.

23 A handshake packet reception handling process 135 is
24 configured to perform handshaking functions and handle

1 packets in the handshaking process. This handshaking
2 process is described below in additional detail.

3 An adjacencies tracking process 140 tracks the nodes
4 105 that are adjacent to the node 105a and programs the
5 nodes in the adjacency set, based upon the handshakes that
6 are performed by the node 105a.

7 The database 115 includes the following data
8 structures: handshake data structure 142, hello time data
9 structure 145, adjacencies data structure 150, and SGK/SID
10 data structure 155.

11 The handshake data structure 142 stores information
12 that determines the nodes 105 that are suitable for
13 handshaking with the node 105a. The handshake packet
14 reception handling process 135 is configured to read data
15 from the handshake data structure 142. The data that are
16 stored in the handshake data structure 142 may include, for
17 example, the values x and y for a one way function $f(x) = y$
18 where y may be a secure hash value if $f(x)$ is a secure hash
19 function, x_1 and x_2 values which may be component values of
20 x , and/or other suitable values.

21 The hello time data structure 145 stores the hello
22 time value (which is the amount of time that needs to
23 expire before the node 105a can initiate another
24 handshaking process). The hello packet process 130 is

1 configured to read data from the hello time data structure
2 145.

3 The adjacencies data structure 150 stores adjacencies
4 information (i.e., nodes 105 that are adjacent to the node
5 105a). The adjacencies tracking process 140 is configured
6 to read data from the adjacencies data structure 150.

7 The SGK/SID values data structure 155 stores all
8 secure group key (SGK) values that are associated with each
9 node 105 that are adjacent to the node 105a. The SGK/SID
10 values data structure 155 also contains secure group
11 identifier (SID) values and values that track all nodes 105
12 that are adjacent to the node 105a. The SGK and SID values
13 are typically bit values.

14

15 The system 100 simplifies the distribution of secure
16 information between nodes 105. Each node 105 in the system
17 100 is programmed with an SID value 165 and SGK value 167,
18 by use of a secure channel (e.g., a console port). The
19 management module 110 can input the programmed SID value
20 165 and SGK value 167 into the secure data structure 175.
21 Once a node 105 is programmed with an SID value and SGK
22 value, the node 105 will no longer require to be programmed
23 with passwords or keys. The node 105 will automatically

1 acquire the passwords or keys from other nodes 105 that are
2 members of a secure group that contains the node 105.

3 Each node (network device) 105 is identified by a
4 unique number, *i*, which could be the Media Access Control
5 (MAC) of the node. As known to those skilled in the art,
6 on a local area network (LAN) or other network, the MAC
7 address is a unique hardware number of a node. When the
8 node is connected to the Internet, a correspondence table
9 relates the Internet Protocol (IP) address of the node to
10 the node's physical (MAC) address on the LAN.

11 The nodes may form a secure group. In the example of
12 Figure 1, the secure group 160 is formed by the nodes 105a,
13 105b, and 105c. Each secure group 150 is identified by the
14 SID 165 and a SGK 167. The secure group 160 is also
15 referred to as the "SID group" 160. Nodes that share the
16 same SID & SGK (i.e., nodes that are in the same secure
17 group) can distribute sensitive or secure information to
18 all other nodes that have the same SID and SGK pair values,
19 as described further below.

20 The security of the system 100 depends on the secrecy
21 of the SGK value. The SGK value, by itself, is not
22 distributed by the nodes in the secure group.
23 Authentication values are generated by sending the SGK
24 value combined with other values by use of a one way

1 function. Typically, this one-way function may be, for
2 example, a one way secure hash. Any suitable hash function
3 may be used in accordance with an embodiment of the
4 inventions. One example of a one-way function is HMAC-MD5
5 (name taken from RFC 3118). As known to those skilled in
6 the art, HMAC (keyed-hash message authentication code) is a
7 type of message authentication code (MAC) calculated using
8 a cryptographic hash function in combination with a secret
9 key. As with any MAC, it may be used to simultaneously
10 verify both the data integrity and the authenticity of a
11 message. Any iterative cryptographic hash function, e.g.,
12 SHA-1, RIPEMD-160, may be used in the calculation of an
13 HMAC; the cryptographic strength of the HMAC depends upon
14 the cryptographic strength of the underlying hash function
15 and on the size and quality of the key.

16 For a one way function that is expressed as, $f(x) = y$,
17 the value y can be easily calculated for a given value of
18 x . However, it would be extremely difficult to calculate
19 the value of x , given the value of y . In one embodiment of
20 the invention, the authentication value y is generated by
21 applying the one way function to the SGK value concatenated
22 with one value selected by each of the nodes that wishes to
23 prove that the node is part of the secure group.

1 Concatenation involves arranging strings of characters into
2 a chained list.

3 In order for two nodes 105 to authenticate each other
4 (determine that both are in the same secure group 160),
5 each node will pass to the other node the two values A_i and
6 B_i , where i is the node number of the node. Once all nodes
7 that wish to establish themselves as members of the secure
8 group (i.e., all authenticating nodes) have received the
9 A_i, B_i values from all of the other authenticating nodes,
10 each of the authenticating node D_1 will perform an
11 authentication operation by use of the one way function.
12 For example, the operation may involve the one way secure
13 hash function, $V = \text{SecureHash}(\text{SGK} + \text{SUM}(A_i \text{ for all } i \text{ where}$
14 $\text{all } i \text{ does not equal } D_1) + B_{D_1})$, where the operation "+" may
15 be an additive operation, a concatenation, an exclusive OR
16 (XOR) or other suitable addition-type operation. In this
17 way, each authenticating node D_1 will generate a value V_i
18 that all of the other authenticating nodes can use to
19 verify that a particular node i knows the SGK value. The
20 node 105 that wishes to prove its membership in the secure
21 group 160 will then send its value V in a multicast.

22 Once two nodes 105 have verified that they are both
23 members of the same secure group 160, then it is safe for
24 them to distribute password information, key information,

1 and/or other secure information between each other, and
2 pairs of nodes will then send each other, for example, a
3 public encryption key for a public key encryption system.
4 Public key encryption is a cryptographic system that uses
5 two keys, which are a public key known to everyone and a
6 private or secret key known only to the recipient of the
7 message. An important element to the public key system is
8 that the public and private keys are related in such a way
9 that only the public key can be used to encrypt messages
10 and only the corresponding private key can be used to
11 decrypt them. Examples of suitable public key encryption
12 system include, for example, the RSA (Rivest-Shamir-
13 Adelman) algorithm, and PGP (pretty good privacy)
14 algorithm.

15 Each node can alternatively encrypt a key for a
16 symmetric encryption system, by using the public key for
17 the node that they have authenticated. A symmetric
18 encryption is a type of encryption where the same key is
19 used to encrypt and decrypt the message. This differs from
20 asymmetric (or public-key) encryption, which uses one key
21 to encrypt a message and another to decrypt the message.
22 Examples of suitable symmetric key encryption system
23 include, for example, the DES single key system and the
24 Rijendael single key system. Once the two nodes have

1 shared the symmetric keys, the two nodes can securely
2 distribute sensitive information between each other for
3 their secure group.

4

5 Example of the Handshaking Process and Encryption Key
6 Establishment

7 In Figure 1, assume that node 105a wishes to prove to
8 node 105b that both node 105a and node 105b are in the same
9 SID group 160 (i.e., secure group 160). Therefore, if node
10 105a and node 105b are in the same SID group 160, then both
11 nodes will have the same SID value 165. Assume that a one
12 way function, $f(x) = y$, will be used in the handshaking
13 process. The handshake process 135 (Figure 1) in the node
14 105a will inform the handshake process (not shown in Figure
15 1) in the node 105b about a value A1 that the handshake
16 process 135 will place in the in the one way function $f(x)$.
17 The value A1 is one of the components for x in the one way
18 function $f(x)$.

19 In response to the A1 value, the handshake process in
20 node 105b will issue a challenge to the handshake process
21 135 in the node 105a. This challenge will be a value B1,
22 which is another component of x in the one way function
23 $f(x)$.

1 The handshake process 135 in node 105a will then
2 appropriately combine the A1 and B1 values of x and
3 calculate $f(x) = y$. The y value is referred to herein as a
4 secure hash value y. It is noted, however, that the value
5 y is not necessarily a secure hash value y because the one
6 way function $f(x) = y$ is not necessarily limited to a
7 secure hash function. When calculating the secure hash
8 value y, the x value also includes the SGK value 167. The
9 handshake process 135 then sends the calculated secure hash
10 value y to the handshake process in node 105b.

11 The A1 value, B1 value, and one way function y value
12 are typically transmitted from one node 105 to another node
13 105 by use of the handshake data packets 152.

14 Note that the SID 165 and SGK 167 are also stored in
15 an SGK/SID values data structure in a database in node
16 105b. If node 105a and node 105b are in the same SID group
17 160, then node 105a and node 105b will have the same values
18 for SID 165 and will have the same values for SGK 167.

19 Node 105b will apply the one way function $f(x)$ for an
20 x value that includes the A1 value, B1 value, and SGK 167
21 to calculate the secure hash value y. Since node 105b and
22 node 105a each has calculated the same secure hash value y,
23 node 105b is now aware that node 105b and node 105a have

1 the same SGK value 167. Therefore, node 105b is now aware
2 that node 105b and node 105a are in the same SID group 160.

3 The node 105b supplies an x component value (B1) as a
4 challenge to node 105a, for use in the calculation of the
5 one way function $f(x)$, so that node 105b can validate that
6 node 105a is a member of the SID group 160. If node 105b
7 does not supply the B1 value as a challenge to node 105a,
8 then node 105a may just use (for the $f(x)$ function
9 calculation) a particular value that it overhears in the
10 network, in order to inform node 105b that it belongs to
11 the same SID group 160. Therefore, the B1 challenge value
12 helps to prevent the vulnerability of the system 100 to a
13 hacker.

14 The node 105a also supplies an x component value (A1)
15 to the $f(x)$ function calculation, so that the one way
16 function is not hacked by a suitable known plain text
17 attack. Therefore, the A1 value provided by node 105a
18 helps to prevent the vulnerability of the system 100 to a
19 hacker.

20 After the above handshaking process has completed,
21 nodes 105a and 105b can establish an encryption key 170 to
22 permit a secure channel of communication between the two
23 nodes 105a and 105b. Therefore, the encryption key 170 is
24 used for secure future communication between the two nodes

1 105a and 105b. As mentioned above, the encryption key 170
2 may be a symmetric encryption key in order to achieve
3 faster processing speed. The encryption key 170 may also
4 be based on the public-private encryption key algorithm,
5 although this may lead to more complexity and slower
6 processing speed.

7 In an embodiment, the encryption key 170 is embedded
8 in the handshake packet that contains the one way function
9 value y. This prevent an unauthorized third party from
10 intercepting the handshake packet with the y value and
11 prevents the unauthorized re-writing of the encryption key
12 170 prior to receipt of the destination node. By embedding
13 the encryption key 170 in the handshake packet with the y
14 value, an unauthorized third party is prevented from re-
15 writing the encryption key 170 unless the third party knows
16 the SGK value 167.

17 In another embodiment, the encryption key 170 may be
18 sent to the destination node as a data packet that is
19 separate from the handshake packet with the y value.

20 The above handshaking process and encryption key
21 establishment is used between other node pairs in the SID
22 group 160. For example, node 105a and node 105c can
23 perform the handshaking process and establish an encryption
24 key for secure communication. As another example, node

1 105b and node 105c can perform the handshaking process and
2 establish an encryption key for secure communication.
3 Other nodes 105 may be included in the SID group 160.
4 Alternatively, the SID group 160 may include only two nodes
5 105.

6 Each node in an SID group 160 form the adjacencies
7 set. The secure group 160 will have stabilized after all
8 adjacencies between nodes 105 in the SID group 160 have
9 been formed by use of the handshaking process.

10 When a password (or other secure information) in the
11 secure data structure 175 is updated in a node 105 (e.g.,
12 node 105a), then the management module 110 will distribute
13 the updated password to the other nodes 105b and 105c in
14 the SID group 160. Therefore, embodiments of the invention
15 advantageously avoid the previous requirement of manually
16 programming each updated password in each node.

17 The system 100 could also be used to distribute keys
18 that permit secure communication across the network. The
19 key should ideally be changed periodically to maintain a
20 secure communication across the network. When a key in the
21 secure data structure 175 is updated, then the management
22 module 110 will distribute the updated key to the other
23 nodes 105b and 105c in the SID group 160.

24

1 Secure Group Construction

2 In order to maintain the security of the system 100,
3 the SGK value 167 must be unknown to other nodes that are
4 not members of the SID group 160. As the distribution
5 increases across the network for the y value (generated by
6 the one way function $f(x)$), the likelihood also increases
7 for an attacker to be able to reverse engineer the SGK
8 value 167. Therefore, it is advantageous to change the SGK
9 value 167 based upon the frequency in which the y value is
10 advertised across the network.

11 Figure 2 is a flowchart of a method 200, in accordance
12 with an embodiment of the invention. In order to prevent
13 attackers from discerning the SGK value 167 by taking
14 numerous samples of hash values y, the number of
15 transmissions of the y value will be limited. Each node
16 105 will only handshake with another node 105 once for
17 every fixed amount of time T (step 205). This time amount
18 T may be programmed into the management module 110. The
19 time amount T or possible ranges for T is user
20 configurable. The time amount T value could start and stop
21 on given dates or could be infinite allowing the user to
22 keep the same keys. The above limitation on the number of
23 handshakes permits a well understood frequency in which the
24 SGK value 167 is changed. For example, assume that only

1 one handshake is performed every time T per node 105 in the
2 network. Since the number of nodes in the SID 160 is known
3 and the time T is pre-determined, then a maximum length of
4 time can be determined before the SGK value 167 will
5 require to be changed. Therefore, the time T serves the
6 purpose of protecting the SGK value 167 by providing a well
7 understood frequency by which the SGK value 167 must be
8 changed to maintain security in the system 100.

9 Each node 105 in the secure group 160 will transmit a
10 HELLO packet for every time (hello time) HT (step 210).
11 The hello time HT could typically be set at a value of
12 approximately one minute. However, hello time HT may also
13 be set to a user configurable value. This HELLO packet
14 will contain the SID value 165, along with the amount of
15 time remaining before that node 105 will be allowed to
16 handshake again. When a particular node 105 detects the
17 presence of a member (node) that is not in the adjacency
18 set (which is stored in adjacencies data structure 150) of
19 that particular node 105, then that particular node 105
20 will attempt to handshake with that member if both the
21 particular node 105 and its new neighbor (the member) have
22 a handshake time remaining value of zero (0) (step 215).
23 By waiting for both nodes to have a remaining value of
24 zero, a reverse attack is prevented, since the above time

1 requirement insures that there is a well understood amount
2 of time that will occur between handshakes and a well
3 understood maximum number of handshakes that will occur per
4 time T.

5 When two nodes 105 discover each other and validate
6 that they are in the same SID group 160, the nodes 105 will
7 share (distribute) their adjacency information for using
8 the VLAN (virtual local area network) (step 220). The
9 adjacency information may be distributed by using the
10 encryption keys 170 that the nodes 105 traded during the
11 handshake process. The adjacency information includes the
12 node IDs and the symmetric key for each node 105. The
13 convergence time for the SID group 160 (i.e., the time when
14 the SID group 160 has stabilized) will be $\text{Log } N$, where N is
15 the number of nodes in the SID group 160.

16 When a particular node 105 obtains the secure
17 adjacency set (information) from another node 105 as a
18 result of the handshaking process, then the particular node
19 105 will redistribute to all of its existing adjacency
20 information to the nodes 105 that it did not previously
21 have in its adjacency set using its own encryption key 170.
22 As noted below, the encryption key 170 may be a symmetric
23 key or a public-private key.

1 The adjacency information will be stored in the
2 adjacencies data structure 150 (Figure 1). The distribution
3 of the adjacency information is distributed on a per VLAN
4 basis. Therefore, the distribution of the adjacency
5 information is performed within the Layer 2 broadcast
6 domain. Each particular node 105 will distribute the
7 adjacency information to adjacent node(s) 105 within the
8 broadcast domain of the particular node 105. In the
9 example of Figure 1, node 105a distributes the adjacency
10 information to adjacent nodes 105b and 105c. Node 105b
11 distributes adjacency information to other nodes that may
12 be adjacent to node 105b. In this way, new nodes 105 will
13 discover each other and establish themselves as members of
14 SID groups 160 and SID groups 160 can be combined.
15 Periodically, for each time period TD, each node 105 will
16 distribute a list of all of its adjacencies information
17 with their symmetric keys 170 to adjacent nodes in the SID
18 group 160 so that all nodes 105 in the SID group 160
19 remains synchronized (step 225).

20

21 Database and Data Distribution

22 All transmission from a node 105 to the secure group
23 160 will typically be transmitted by using the encryption

1 key 170 (e.g., symmetric key) for that node 105 and will be
2 sent to a Secure Group Management multicast MAC.

3 The secure group 160 will be used for distributing
4 sensitive information that is specific to that secure group
5 160. Each node 105 maintains a database of information
6 that can be queried by authorized software entities that
7 are running on the node 105. For example, the management
8 module 110 can query the secure data structure 175 that
9 stores the sensitive information. As shown in Figure 3, in
10 an embodiment, the secure data structure 175 includes a
11 table 300 with a field name column 305, field data column
12 310, and revision information column 315. The field name
13 column 305 identifies the type of database entry that can
14 be queried by the management module 110. For example, the
15 field names may include password 320 which indicates that
16 the data in field 325 is the password information that is
17 used by the secure group 160. The field 330 indicates
18 revision information for the password information in field
19 325.

20 As another example, the field name column 300 may
21 include key 335 which indicates that the data in field 340
22 is a key value that is used for secure communication
23 between the nodes 105 in the secure group 160. The field

1 345 indicates the revision information for the key value in
2 field 340.

3 As another example, the field name column 300 may
4 include other secure information 350 which indicates that
5 the data in field 355 is another type of secure information
6 that is relevant for the secure group 160.

7 The revision information in fields 330 or 345 may
8 indicate, for example, a sequence number of the associated
9 secure data in field 325 or 340, respectively. The
10 sequence number is incremented for each update occurrence
11 in the secure data.

12 The revision information in fields 330 or 345 may
13 alternatively indicate, for example, a modification date of
14 the associated secure data in field 325 or 340,
15 respectively. The modification date will indicate when the
16 secure data was previously updated.

17 The revision information in fields 330 or 345 may
18 alternatively indicate, for example, a delta time (DTV)
19 indicating the elapsed time since the previous modification
20 of the associated secure data in field 325 or 340,
21 respectively. An advantage of using the delta time value
22 for the revision information column 315, rather than using
23 a date of last modification, is that using a date of last
24 modification requires that systems have a secure mechanism

1 for maintaining synchronized system clocks. The advantage
2 of using the DTV value over a sequence number is that
3 sequence numbers could have problems in determining the
4 value that is actually the most recent version of the data,
5 when the SID group 160 becomes disjoint (discontiguous).
6 An SID group 160 can become disjoint, for example, when the
7 SID group 160 is being constructed or after a network
8 topology change.

9

10 The revision information in column 315 insures that
11 each node 105 in the secure group 160 is able to receive
12 and maintain the latest version of the secure information
13 in the secure data structure 175. The revision information
14 in column 315 determines an age of the associated secure
15 information so that each node 105 in the secure group 160
16 will store a latest version of the secure information.

17

18 In addition, the node 105 will update all adjacent
19 nodes 105 when the database (table 300) of the node 105 is
20 modified. As a result, all of the database for a
21 particular SID group 160 will be kept synchronized. When
22 the table 300 is modified internally, the node 105 will
23 redistribute the modified entries of table 300 to all of
24 its adjacent nodes 105 that are members of the SID group

1 160. This redistribution may occur either immediately or
2 after a specified wait interval so that the modifications
3 are lumped together. When a particular node 105 receives
4 an entry update from an adjacent node 105 that changes the
5 state of its database table 300, the particular node 105
6 will redistribute that change to all of its adjacent
7 neighbor nodes 105 (excluding the inbound VLAN) with a
8 single set of multicast frames for each VLAN that the
9 particular node 105 is running on. For every time period
10 TD, the particular node 105 will distribute its database
11 table 300 changes to its adjacent neighbor nodes 105.

12 As mentioned above, the secure database system 100 can
13 be used to distribute any sensitive information. In one
14 embodiment, the secure database system 100 can be used to
15 distribute password information to nodes 105 that are
16 switches, so that a user can update their password for all
17 switches on the network by having the modified password
18 propagate across the network. The secure database system
19 100 could also be used as part of a key management system
20 for applications that require keys to be shared between
21 nodes 105.

22

1 Resolving Ambiguity Between Entry Updates

2 In the event that a particular node 105 receives an
3 update for an entry (in the table 300), where the updated
4 entry has the same sequence value (this sequence value
5 could be a version number or a date) as the entry that the
6 particular node 105 already has in its database table 300,
7 the particular node 105 will pick the entry that has the
8 larger data value and discard the other entry, in an
9 embodiment of the invention. As a result, the database
10 tables 300 in the nodes 105 of the secure group 160 will
11 remain synchronized even if they receive different data
12 elements with the same sequence, version, and date
13 information.

14

15 Figure 4 illustrates an example of a method to resolve
16 ambiguity between entry updates. Assume that the
17 management module 110 detects that the currently stored
18 secure information 325a and the recently received updated
19 secure information 325b. The recently received updated
20 secure information 325b is typically buffered into, for
21 example, an inbound packet buffer 405. Assume that both
22 secure information 325a and 325b have the same revision
23 information 315 (e.g., both of the secure information 325a
24 and 325b have the same sequence number). If the management

1 module 110 determines that secure information 325a is
2 larger in data value than the secure information 325b, then
3 the management module 110 will keep the secure information
4 325a as stored in the secure data structure 175 and will
5 discard the recently received secure information 325b. On
6 the other hand, if the management module 110 determines
7 that secure information 325b is larger in data value than
8 the secure information 325a, then the management module 110
9 will store the secure information 325b into the secure data
10 structure 175 and will discard the secure information 325a.

11 The above condition of ambiguity between entry updates
12 in the tables 300 typically occur, for example, if sequence
13 numbers are being used for the revision information column
14 315 and the network becomes temporarily discontinuous, or
15 if the values in the table 300 are being updated too
16 quickly.

17

18 Protecting SGK Values and Symmetric Keys

19 Figure 5 is a block diagram illustrating a method 500
20 for increasing the security of the secure group. As the
21 number of nodes 105 in a particular SID group 160
22 increases, the system 100 may become more vulnerable to an
23 attack. With each additional node 105, the number of
24 sample hash values y that an attacker can acquire per unit

1 time increases. The number times that the y value is
2 advertised will influence the amount of time that may pass
3 before the SGK 167 for an SID 165 must be changed. In
4 order to compensate for an increased number of nodes 105 in
5 an SID group 160, the SGK value 167 is widened, the output
6 of the one way function $f(x)$ will be larger, the symmetric
7 keys 170 will be larger, and/or the times T, TD, and HT
8 will be increased. The security of the system 100 can be
9 increased by widening the SGK value 167 (step 510). For
10 example, the SGK value 167 can be widened from a few
11 hundred bits to a few thousand bits.

12 In addition to sharing adjacency information (in data
13 structure 150 in Figure 1) and database information (in
14 secure data structure 175), each node 105 will also have to
15 generate new symmetric keys 170 periodically. The amount
16 of time between symmetric key regeneration, TK, can be
17 decreased (step 515) to increase the security of the system
18 100. By periodically generating symmetric keys 170 with
19 new values, an attacker is unlikely to determine the values
20 of the symmetric keys 170.

21

22 Password Distribution

23 In an embodiment of the invention, passwords will be
24 administered by using a command that updates one of the

1 database tables 300 in the nodes 105. The management
2 module 110 (Figure 1) may be used to update the table 300
3 with the password data in field 325. This update will
4 cause the new password information to be distributed to all
5 nodes 105 that are members of a particular secure group
6 160.

7

8 Rapid Convergence

9 In order to allow for rapid group construction (such
10 as when a plurality of nodes 105 are booted), the nodes 105
11 may be configured to transmit a burst of NB handshakes for
12 every amount of time TB. NB is the number of handshakes
13 and TB is the time amount between burst of handshakes.
14 Note that the time T, as mentioned above, is the time
15 between handshakes. The values of TB and NB can be set via
16 the management module 110.

17

18 Avoiding Excessive Joins

19 To prevent a single node 105 from attempting to
20 handshake with numerous adjacent nodes after booting or
21 after two SID groups 160 are joined through a network
22 topology change, each node 105 will only try to establish
23 membership with one adjacent node at a time, and will wait
24 at time $TW \pm TR$ between handshake attempts, where TW is a

1 fixed configurable time amount and TR is a random amount of
2 time that is bounded by a bound range that can be specified
3 by the user. The TW and TR values are set to any suitable
4 values so that the nodes do not attempt to communicate at
5 the same time. A variance of approximately two seconds to
6 approximately one minute for the TW and TR values would be
7 acceptable.

8 For example, node 105a will only try to establish
9 membership in the SID group 160 with node 105b, and another
10 node (e.g., node 105c) will try to establish membership in
11 the SID group 160 with another node (e.g., node 105b).
12 Node 105a can then establish membership in the SID group
13 160 with another node (e.g., node 105c).

14

15 Therefore, embodiments of the invention simplify the
16 distribution of secure information between nodes. For
17 example, embodiments of the invention simplify the
18 administration of network passwords, the distribution of
19 key information, and/or the distribution of other types of
20 secure information.

21

22 The various engines, tools, or modules discussed
23 herein may be, for example, software, firmware, commands,

1 data files, programs, code, instructions, or the like, and
2 may also include suitable mechanisms.

3 Reference throughout this specification to "one
4 embodiment", "an embodiment", or "a specific embodiment"
5 means that a particular feature, structure, or
6 characteristic described in connection with the embodiment
7 is included in at least one embodiment of the present
8 invention. Thus, the appearances of the phrases "in one
9 embodiment", "in an embodiment", or "in a specific
10 embodiment" in various places throughout this specification
11 are not necessarily all referring to the same embodiment.
12 Furthermore, the particular features, structures, or
13 characteristics may be combined in any suitable manner in
14 one or more embodiments.

15 Other variations and modifications of the above-
16 described embodiments and methods are possible in light of
17 the foregoing disclosure. Further, at least some of the
18 components of an embodiment of the invention may be
19 implemented by using a programmed general purpose digital
20 computer, by using application specific integrated
21 circuits, programmable logic devices, or field programmable
22 gate arrays, or by using a network of interconnected
23 components and circuits. Connections may be wired,
24 wireless, by modem, and the like.

1 It will also be appreciated that one or more of the
2 elements depicted in the drawings/figures can also be
3 implemented in a more separated or integrated manner, or
4 even removed or rendered as inoperable in certain cases, as
5 is useful in accordance with a particular application.

6 It is also within the scope of an embodiment of the
7 present invention to implement a program or code that can
8 be stored in a machine-readable medium to permit a computer
9 to perform any of the methods described above.

10 Additionally, the signal arrows in the
11 drawings/Figures are considered as exemplary and are not
12 limiting, unless otherwise specifically noted.
13 Furthermore, the term "or" as used in this disclosure is
14 generally intended to mean "and/or" unless otherwise
15 indicated. Combinations of components or steps will also
16 be considered as being noted, where terminology is foreseen
17 as rendering the ability to separate or combine is unclear.

18 As used in the description herein and throughout the
19 claims that follow, "a", "an", and "the" includes plural
20 references unless the context clearly dictates otherwise.
21 Also, as used in the description herein and throughout the
22 claims that follow, the meaning of "in" includes "in" and
23 "on" unless the context clearly dictates otherwise.

1 It is also noted that the various functions,
2 variables, or other parameters shown in the drawings and
3 discussed in the text have been given particular names for
4 purposes of identification. However, the function names,
5 variable names, or other parameter names are only provided
6 as some possible examples to identify the functions,
7 variables, or other parameters. Other function names,
8 variable names, or parameter names may be used to identify
9 the functions, variables, or parameters shown in the
10 drawings and discussed in the text.

11

12 The above description of illustrated embodiments of
13 the invention, including what is described in the Abstract,
14 is not intended to be exhaustive or to limit the invention
15 to the precise forms disclosed. While specific embodiments
16 of, and examples for, the invention are described herein
17 for illustrative purposes, various equivalent modifications
18 are possible within the scope of the invention, as those
19 skilled in the relevant art will recognize.

20 These modifications can be made to the invention in
21 light of the above detailed description. The terms used in
22 the following claims should not be construed to limit the
23 invention to the specific embodiments disclosed in the
24 specification and the claims. Rather, the scope of the

1 invention is to be determined entirely by the following
2 claims, which are to be construed in accordance with
3 established doctrines of claim interpretation.